Available online at http://www.ijasrd.org/in

International Journal of Advanced Scientific Research & Development

e-ISSN: 2395-6089
p-ISSN: 2394-8906

Vol. 02, Iss. 03, Ver. II, Jul – Sep' 2015, pp. 128 – 134

# GENERATING SENTIMENT PROFILE OF MOVIES FROM UNSTRUCTURED TEXTUAL REVIEWS

**Dr. Vivek Kumar Singh**

Asst. Prof., Dept., of Computer Science, Banaras Hindu University, Varanasi – 221 005, India.

**ARTICLE INFO**

***Article History:***

Received: 20 Sep 2015;
Received in revised form: 23 Sep 2015;
Accepted: 23 Sep 2015;
Published online: 30 Sep 2015.

***Key words:***

Movie Reviews,
Opinion Mining,
Sentiment Analysis

**ABSTRACT**

This paper presents an experimental framework for computing opinion polarity summary from free-form unstructured textual reviews of a movie. The system works on movie reviews obtained from the World Wide Web and computes a sentiment polarity summary in an automated manner. The experimental design is based on use of the SentiWordNet Lexicon. Two labeled movie review datasets are used to evaluate the design. The standard performance evaluation metrics of accuracy, f-measure and entropy are calculated to demonstrate the suitability of the proposed framework.



## INTRODUCTION

Sentiment analysis is language processing task that uses an algorithmic formulation to identify opinionated content and categorize it as having "positive", "negative" or "neutral" polarity. It has been formally defined as an approach that works on a quintuple $< O_i, F_{ij}, S_{kijl}, H_k, T_l >$; where, $O_i$ is the target object, $F_{ij}$ is a feature of the object $O_i$, $S_{kijl}$ is the sentiment polarity (+ve, -ve or neutral) of opinion of holder $k$ on $j^{th}$ feature of object $i$ at time $l$, and $T_l$ is the time when the opinion is expressed (Liu, 2009).

Sentiment analysis is now a very useful task across a wide variety of domains. Whether it is commercial exploitation by organizations for identifying opinions about products/services from various customers, or identifying the election prospect of political candidates; sentiment analysis finds its applications. The user created information is very useful for companies which try to know the feedback about their products or services. This feedback helps them in taking informed decisions. However, the large number of reviews becomes information overload. Sentiment analysis fills this gap by producing a sentiment profile computed from a large number of user reviews about a product or service. A user is not required to read all the reviews of a product/service but s/he can have access to an overall sentiment profile which tells how many reviews are positive and how many are negative.

This paper presents a SentiWordNet based approach for sentiment classification of movie reviews. The document-level sentiment classification formulation designed utilizes diverse linguistic features ranging from adjectives to verbs. Two schemes, AAC (Adverb Adjective Combination) and AAAVC (Adverb Adjective Adverb Verb Combination) are explored for sentiment analysis task. The rest of the paper is organized as follows. Section 2 describes the task and method of Sentiment Analysis. Section 3 presents the dataset used and implementation. Section 4 describes the results obtained and the paper concludes in section 5.

## SENTIMENT ANALYSIS

There are mainly three type of techniques for sentiment classification of subjective texts: (a) using a machine learning based text classifier such as Naïve Bayes, SVM or KNN; (b) using semantic orientation schemes of extracting relevant n-grams of the text and then labeling them either as positive or negative and consequently as the document; and (c) using the SentiWordNet based publicly available lexicon dictionary that provides positive, negative and neutral scores for the words.

### 2.1 Naïve Bayes Implementation

It is a supervised probabilistic machine learning classifier. The sentiment analysis problem can be visualized as 2-class text classification problem that is classifying text documents in two polarity classes, namely “positive” polarity and “negative” polarity. In Naive Bayes classifier, the probability of document $d$ being in class $c$ is computed as.

$$P(c|d) \propto P(c) \prod_{1 \leq k \leq d} P(w_k \mid c)$$

where, the term $P(c)$ refer to prior probability of a document occurring in class $c$ and corresponds to the majority class. The expression $P(w_k \mid c)$ is the conditional probability of a term $w_k$ occurring in a document of class $c$. The term $P(w_k \mid c)$ is interpreted to be the measure of how much evidence the term $w_k$ contributes that $c$ is correct class. The main idea in this classification is to classify the document based on statistical pattern of occurrence of terms. The goal in text classification using Naive Bayes is to determine the best class for a document. The best class in Naive Bayes classification is the maximum a-posteriori (MAP) class, computed as:

$$c_{map} = \arg\max_{c \in C} P(c|d) = \arg\max_{c \in C} P(c) \prod_{1 \leq k \leq n_d} P(w_k \mid c)$$

The probability values are computed from the training set. The multiplication of many conditional probability terms can be reduced by adding logarithms of probabilities. Therefore, above equation can be modified as:

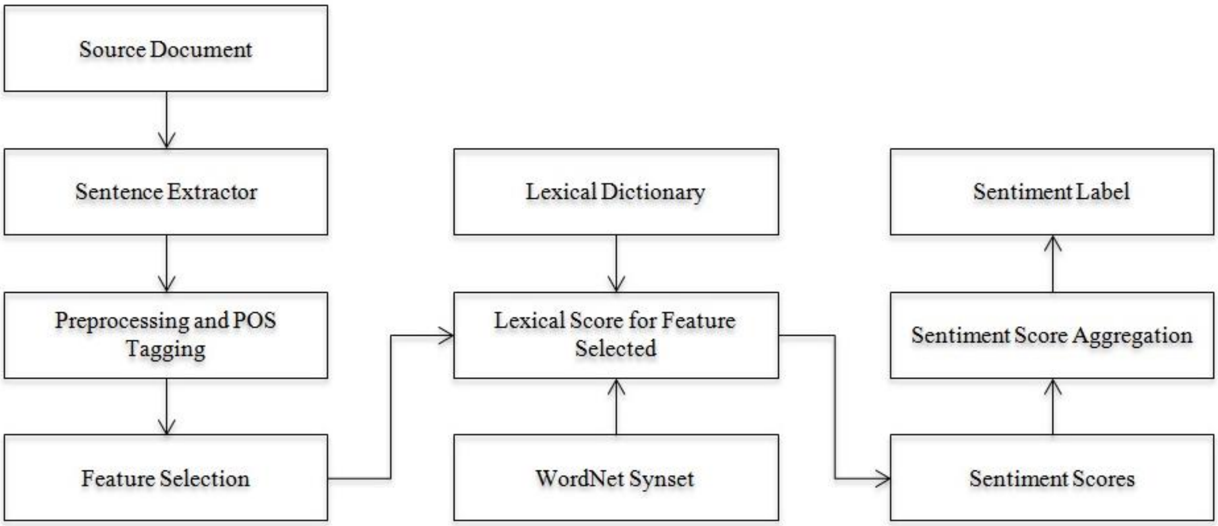
$$c_{map} = \arg\max_{c \in C} [\log P(c) + \sum_{1 \leq k \leq n_d} P(w_k \mid c)]$$

In equation above, each $P(w_k \mid c)$ term refer to the weight which specify how good an indicator the term $w_k$ is for class $c$, and in similar way the prior $\log P(c)$ indicates the relative frequency of class $c$ (Manning et al., 2008).

**2.2 Sentiwordnet Approaches**

Two different methods based on of SentiWordNet lexicon are implemented here. These methods vary in term of their feature selection, weight assignment and aggregation schemes. In the first implementation, every document (that was tagged with POS tags) is parsed to extract all terms with POS tag "adjectives". Based on reported results in (Chesley, 2005), (Karamibekr & Ghorbani, 2012), it is known that adverbs and verbs play an important role in sentiment analysis. Therefore, this paper tried to use "adverbs" and "adjectives" as features in implementations. In the second version, "adverb+adjective" and "adverb+verb" combines are used as features. The basic steps and information flow in the SentiWordNet-based methods is shown in Figure 1.

**Figure 1:** *Lexicon Based Approach General Architecture*




For each extracted term, SentiWordNet lexicon is scanned and the two scores (positive polarity and negative polarity strengths weighted over a number of synsets) are obtained. An expanded scan is performed by including the synonyms (obtained from WordNet) of the term, if the original term does not occur in the SentiWordNet lexicon. Hereafter, the two versions implemented are referred to as SWN (AAC) and SWN (AAAVC). Since "adverbs" modify the scores of succeeding terms, it needs to be decided as to what proportion the sentiment score of an "adjective" or a "verb" should be modified by the preceding "adverb". The modifying weight (scaling factor) of adverb score is taken as 0.35 based on the conclusions reported in (Karamibekr & Ghorbani, 2012). The other main issue that remains to be addressed is how should the sentiment scores of extracted "adverb+adjective" and "adverb+verb" combines, in a sentence of the document, should be aggregated. If sentiment polarity score total of an "adverb+adjective" combine is "x" and "adverb+verb" combine is "y"; then the net sentiment score of these two taken together will be $x + 0.3y$, if the weightage factor for "adverb+verb" combine is 30 percent. The indicative pseudo-code for the SWN (AAC) and SWN (AAAVC) methods are given below, in Algorithm 1 and 2, respectively. A careful look at the steps will help to understand the manner and rationale of the expressions. A more detailed discussion on SentiWordNet based implementations are reported in (Singh et al, 2013a, 2013b, 2013c). All implementations have been done in JAVA with NetBeans IDE.

**Figure 2:** *Pseudocode of SWN (AAC) Algorithm*

**Algorithm 1** Algorithm SWN (AAC)

```
For each sentence, extract adv+adj combines.
For each extracted adv+adj combine do:
if score(adj) = 0 then
    ignore it.
end if
if score(adv) > 0 then
    if score(adj) > 0 then
        fs_AAC(adv, adj) = min(1, score(adj) + sf * score(adv))
    end if
    if score(adj) < 0 then
        fs_AAC(adv, adj) = min(1, score(adj) − sf * score(adv))
    end if
end if
if score(adv) < 0 then
    if score(adj) > 0 then
        fs_AAC(adv, adj) = max(−1, score(adj) + sf * score(adv))
    end if
    if score(adj) < 0 then
        fs_AAC(adv, adj) = max(−1, score(adj) − sf * score(adv))
    end if
end if
Add the postive and negative scores to respective pools.
```

**Figure 3:** *Pseudocode of SWN (AAC) Algorithm*

**Algorithm 2** Algorithm SWN (AAAVC)

```
For each sentence, extract adv+adj and adv+verb combines.
For each extracted adv+adj combine do:
if score(adj) = 0 then
    ignore it.
end if
if score(adv) > 0 then
    if score(adj) > 0 then
        f(adv, adj) = min(1, score(adj) + sf * score(adv))
    end if
    if score(adj) < 0 then
        f(adv, adj) = min(1, score(adj) − sf * score(adv))
    end if
end if
if score(adv) < 0 then
    if score(adj) > 0 then
        f(adv, adj) = max(−1, score(adj) + sf * score(adv))
    end if
    if score(adj) < 0 then
        f(adv, adj) = max(−1, score(adj) − sf * score(adv))
    end if
end if
For each extracted adv+verb combine do:
if score(verb) = 0 then
    ignore it.
end if
if score(adv) > 0 then
    if score(verb) > 0 then
        f(adv, verb) = min(1, score(verb) + sf * score(adv))
    end if
    if score(verb) < 0 then
        f(adv, verb) = min(1, score(verb) − sf * score(adv))
    end if
end if
if score(adv) < 0 then
    if score(verb) > 0 then
        f(adv, verb) = max(−1, score(verb) + sf * score(adv))
    end if
    if score(verb) < 0 then
        f(adv, verb) = max(−1, score(verb) − sf * score(adv))
    end if
end if
Add the postive and negative scores to respective pools.
```

## DATASET

The experimental frameworks are implemented and run on two datasets collected. The datasets include two review sets of 800 reviews for Hindi and 800 for English movies. There are 10 reviews each for 80 Bollywood and 80 Hollywood movies from the IMDB, a popular movie review website. Table 1 describes the detail of datasets.

**Table 1:** *Details of Dataset Used*

| Dataset | No. of Reviews | Avg. Length (in Words) |
|---|---|---|
| Bollwood Movies | 800 | 323 |
| Hollywood Movies | 800 | 300 |

## RESULT

The document-level sentiment analysis results on two datasets using two different implementations (SWN (AAC) and SWN (AAAVC)). The table 2 presents the Accuracy F-Measure and Entropy values with two implementations. The result shows that SWN (AAAVC) achieve better accuracy than SWN (AAC). The table 3 presents category wise accuracy percentage of movie reviews labeled as 'positive' or 'negative' by different methods. The table 4 presents the total number of 'positive' or 'negative' assigned by these two methods. The figure 4 presents results summary of SWN implementations. Experimental results are also obtained for Naïve Bayes implementation. The table 5 and figure 5 present results of Naïve Bayes implementation for the two datasets. Results with varying training data are computed and it can be clearly seen that higher training data results in higher accuracy.

**Table 2:** *Performance Result on three Different Dataset*

| Methods | Datasets | | |
|---|---|---|---|
| | | Dataset1 | Dataset2 |
| SWN(AAC) | Accuracy | 76.0% | 74.75% |
| | F-Measure | 0.6943 | 0.6895 |
| | Entropy | 0.2174 | 0.2393 |
| SWN(AAAVC) | Accuracy | 76.50% | 75.88% |
| | F-Measure | 0.6994 | 0.6894 |
| | Entropy | 0.2032 | 0.2333 |

**Table 3:** *Category wise Accuracy Percentage Assigned by Two Approaches*

| Methods | Datasets | | |
|---|---|---|---|
| | | Dataset1 | Dataset2 |
| SWN(AAC) | Positive | 91.11% | 81.94% |
| | Negative | 41.63% | 56.25% |
| SWN(AAAVC) | Positive | 94.36% | 85.76% |
| | Negative | 37.95% | 50.44% |

**Table 4:** *Total Number of Positive and Negative Labels Assigned by Two Approaches*

| **Methods** | **Datasets** | | |
|---|---|---|---|
| | | **Dataset1** | **Dataset2** |
| **SWN(AAC)** | Positive | 506/555 | 472/576 |
| | Negative | 102/245 | 126/224 |
| **SWN(AAAVC)** | Positive | 519/550 | 494/576 |
| | Negative | 93/245 | 113/224 |

**Figure 4:** *Accuracy Plot for Different Datasets*

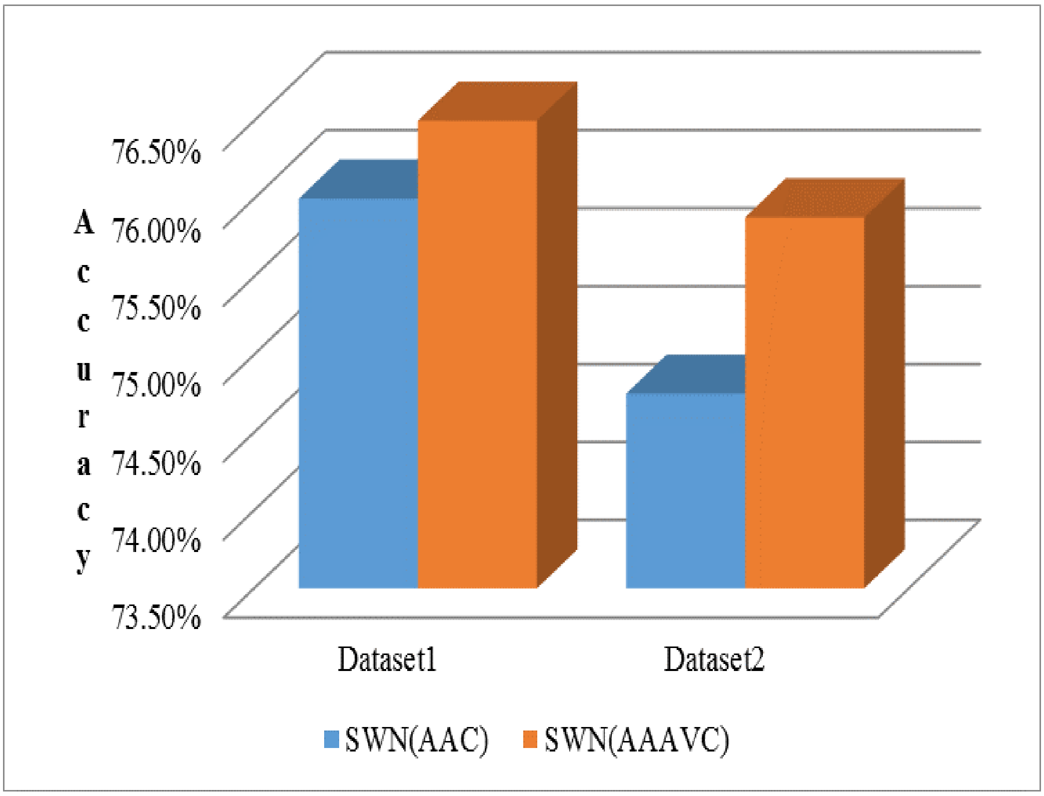
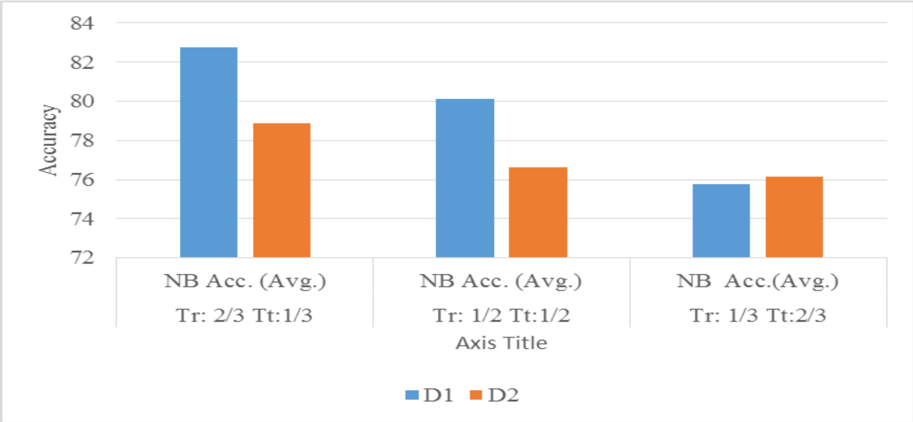



**Table 5:** *Performance Result on Two Different Dataset*

| Training Size | Run | D1 | D2 |
|---|---|---|---|
| Train: ½<br>Test: ½ | Train Doc. (p : n) | 277:122 | 288:112 |
| | NB Acc. (Avg.) | 80.123 | 76.62 |
| Train: 1/3<br>Test: 2/3 | Train Doc.(p: n) | 185:81 | 192:74 |
| | NB Acc.(Avg.) | 75.744 | 76.126 |
| Train: 2/3<br>Test: 1/3 | Train Doc. (p : n) | 370:162 | 384:148 |
| | NB Acc. (Avg.) | 82.750 | 78.871 |

**Figure 5:** *Accuracy Plot for Different Datasets*




## CONCLUSION

The paper presents experimental evaluation of SentiWordNet based and Naïve Bayes approaches for sentiment analysis. The two algorithmic implementations are implemented and evaluated on two movie datasets. Detailed results of accuracy of classification of both the approaches are presented. It is observed that accuracy levels obtained are reasonably good. These implementations can thus be used to generate a sentiment profile summary from reviews of a movie. The approaches can also be used with other review type data.

## REFERENCES

[1] Chesley, P., Vincent, B., Xu, L. and Srihari, R.K. (2006). Using Verbs and Adjectives to Automatically Classify Blog Sentiment. American Association for Artificial Intelligence.

[2] Karamibekr, M., and Ghorbani, A. A. (2012). Verb Oriented Sentiment Classification. Proceedings of International Conference on web Intelligence and Intelligent Agent Technology, 327-331.

[3] Liu, B. (2009). Sentiment Analysis And Opinion Mining. Proceedings of 5th Text Analytics Summit, Boston.

[4] Manning, C. D., Raghavan, P., and Schutze, H. (2008). Introduction to Information Retrieval. Cambridge University Press, New York, USA.

[5] Singh, V. K., Piryani, R., Uddin, A., & Waila, P. (2013a). Sentiment analysis of textual reviews; Evaluating machine learning, unsupervised and SentiWordNet approaches. In Proceedings of 5th International Conference on Knowledge and Smart Technology (KST), 122-127.

[6] Singh, V. K., Piryani, R., Uddin, A., & Waila, P. (2013b). Sentiment analysis of Movie reviews and Blog posts. In Proceedings of 3rd International Conference on Advance Computing Conference, 893-898.

[7] Singh, V. K., Piryani, R., Uddin, A., & Waila, P. (2013c). Sentiment analysis of movie reviews: A new feature-based heuristic for aspect-level sentiment classification. In Proceedings of International Multi-Conference on Automation, Computing, Communication, Control and Compressed Sensing, 712-717.